

# IIA - Progetti relativi ad AI simbolica e Knowledge Graphs

## Recommended 2024-2025

- **Integrated Music Knowledge Base (IMKB) - base per progetti indicati sotto**
  - Di seguito si illustra un macro-tema, all'interno del quale sono definiti progetti specifici, che possono essere sviluppati indipendentemente. Se lo vorranno, i gruppi potranno interagire o addirittura collaborare su aspetti comuni. Ciascun gruppo discuterà, i propri risultati.
  - Obiettivo principale: familiarizzare con il problema di costruire basi di conoscenza a partire da sorgenti diverse, usando i knowledge graph (KG) sia come sorgente sia come modello integrato della conoscenza.
  - Dominio: musica.
  - Idea di fondo: arricchire dati musicali estratti da sorgenti non semantiche (playlist, ed eventualmente canzoni) con dati provenienti da KG relativi soprattutto agli artisti; usare i dati arricchiti per elaborazioni di vario tipo non possibili sui dati di partenza
  - Knowledge Graph di riferimento: Wikidata - ma è possibile passare a DBpedia se si riscontrano difficoltà (DBpedia ha un'ontologia più pulita e facile da comprendere).
  - Altre sorgenti da considerare per l'integrazione:
    - Playlist (raccomandato):
      - <https://www.aicrowd.com/challenges/spotify-million-playlist-dataset-challenge>
    - Canzoni e diverse caratteristiche (tag su generi musicali, dati utenti, violenza, etc.)
      - <http://millionsongdataset.com/>
      - <https://www.kaggle.com/datasets/saurabhshahane/music-dataset-1950-to-2019>
  - Altre risorse utili per il progetto:
    - Mapping concettuale tra Wikidata e risorse in diverse sorgenti:
      - [https://www.wikidata.org/wiki/Wikidata:WikiProject\\_Music](https://www.wikidata.org/wiki/Wikidata:WikiProject_Music)
  - Consiglio: lavorate su sottoinsiemi manipolabili, possibilmente selezionati con un criterio ragionevole
- Note:
  - I gruppi possono proporre e implementare variazioni su quanto suggerito (, mantenendo gli obiettivi di base
- **Progetto Ontologia - 1 Persona [IMKB-ONTO]**
  - Modellare un'ontologia musicale che descrive gli artisti e le loro proprietà, partendo dalle classi di Wikidata apportando eventualmente le modifiche necessarie
  - Modellare l'ontologia in OWL in Protegè usando almeno tutti i costrutti RDFS, e, possibilmente, costrutti aggiuntivi di OWL

- Definire esempi di inferenze utili per un'utente, esemplificando interrogazioni a cui non è possibile rispondere senza inferenze
- Persistere la base di conoscenza in un triple store che supporti interrogazioni.
- Utilizzare un reasoner per calcolare le inferenze, materializzare le inferenze e mostrare l'effetto sulle query definite.
- Output:
  - software/risultati: base di conoscenza, ontologia OWL, query usate per testare il lavoro
  - Relazione scritta - data di consegne
  - Presentazione Power Point - giorno dell'esame
- Il voto terrà conto della dimensione della base di conoscenza (numero di triple e inferenze), della metodologia adottata per la specifica dell'ontologia (appropriatezza delle scelte fatte e complessità delle inferenze), nonché della capacità di argomentazione.
- **Progetto Integrazione con Focus su Disambiguazione Artisti - 1 Persona [IMKB-Integrazione]**
  - Integrare i dati relativi alle playlist (o, nel caso peggiore, alle canzoni) cercando di disambiguare i nomi degli artisti nelle playlist con Wikidata; a tale scopo è possibile usare algoritmi definiti dallo studente, servizi esistenti (ad es., sistemi di Named Entity Linking - anche se il contesto limitato potrebbe limitarne l'efficacia), LLM, o combinazioni di questi strumenti
  - Persistere la base di conoscenza integrata in un triple store e mostrare risultati di interrogazioni che non sarebbe possibile ottenere senza integrazione
  - Volendo è possibile definire una piccola ontologia per la base di conoscenza integrata, per lo più per modellare i dati nella base di conoscenza integrata
  - Valutare punti di forza e limiti di quanto fatto, calcolando anche Precision / Recall / F-Measure del task di integrazione su un insieme di dati campione.
  - Output:
    - software/risultati: base di conoscenza integrata, query usate per testare il lavoro
    - Relazione scritta - data di consegne
    - Presentazione Power Point - giorno dell'esame
  - Il voto terrà conto della dimensione della base di conoscenza (numero di triple e link), della metodologia adottata per la riconciliazione delle entità (appropriatezza delle scelte fatte), nonché della capacità di argomentazione.
- **Progetto Integrazione Applicativa - 2 persone [IMKB-APP]**
  - Sviluppare un'applicazione web che permette di embeddare una playlist in una pagina e cercare ulteriori informazioni relative agli artisti che appaiono nella playlist e, eventualmente, alle canzoni contenute nella playlist
  - Nel fare questo, si affronti il problema di collegare i dati nelle playlist con gli artisti di Wikidata, utilizzando metodi ad hoc o servizi esterni. Tali servizi devono essere eseguiti in modalità live, sulla base della playlist caricata dall'utente.
  - Per arricchire i dati delle playlist si scelgano proprietà che possano recuperare informazioni tali da fornire interazioni inusuali con i dati (ad es., luogo di nascita degli artisti con display su una mappa, data della canzone

con timeline, etc) e si sviluppi almeno qualche funzionalità di interazione con tali dati.

- Bonus: utilizzare le informazioni arricchite per implementare un meccanismo primitivo di raccomandazione di contenuti simili.
- Si discutano limiti dell'applicazione e possibili miglioramenti
- Output:
  - software/risultati: applicazione web (su portatile studenti) con pagina in cui caricare la playlist e visualizzazione
  - Relazione scritta - data di consegne
  - Presentazione Power Point - giorno dell'esame
- Il voto terrà conto della dimensione della base di conoscenza e rilevanza delle nuove interrogazioni (numero di nuove triple per playlists etc ), della metodologia adottata per integrare la conoscenza (appropriatezza delle scelte fatte), della qualità finale del servizio realizzato, nonché della capacità di argomentazione.

## Progetti 2023-2024 ancora disponibili

- **Costruzione di KG con ecosistema RML.IO (progetti anno passato ancora disponibili)**
  - Il progetto consiste nel testare RML (<https://rml.io/>) come linguaggio di mapping da dati tabellari a RDF. RML è un linguaggio con una specifica (<https://rml.io/specs/rml/>), ma è associato a diverse applicazioni per editare i mapping (si veda <https://rml.io/> per maggiori dettagli), ed eseguire la trasformazione (ad esempio RMLMapper - <https://github.com/RMLio/rmlmapper-java> - in JAVA o RMLStreamer - <https://github.com/RMLio/RMLStreamer> - in Scala, preferibile). Lasciando allo studente la scelta di eventuali strumenti di appoggio il progetto consiste in due parti, che verranno entrambe presentate :
    - Sintetizzare le principali caratteristiche di RML in una presentazione power point
    - Definire le trasformazioni per un file CSV a scelta (in caso di dubbi contattare il docente e verrà fornito anche il CSV), ed eseguirle mediante uno dei tool.
  - In alternativa ai riferimenti sopra, è anche possibile usare l'approccio MORPH, sempre basato su RML
    - <https://github.com/morph-kgc/morph-kgc/>
  - Il progetto tratta un argomento molto simile a "Costruzione di KG con MORPH-KGC"; gli studenti possono confrontarsi per capire assieme le parti comuni.
- **Ragionare su Knowledge Graph in Prolog**
  - Il progetto consiste nel riprodurre meccanismi inferenziali per knowledge graph in Prolog. La proposta è di partire dalle inferenze supportate da linguaggi nativi come RDFS e OWL e replicare i pattern inferenziali in Prolog. Si suggerisce di partire da RDFS e poi aggiungere livelli ulteriori di inferenza incrementalmente a partire da OWL. Nel progetto è possibile prendere ispirazione da alcuni spunti; ad esempio, il sottoinsieme di OWL OWL-RL è pensato proprio per poter essere implementato in sistemi a regole; RDF++ è

usato in motori inferenziali in produzione come superinsieme di RDFS particolarmente interessante. Il progetto deve appoggiarsi a un dataset di esempio in RDF, in maniera tale da esemplificare le inferenze ottenute mediante le regole. Ad esempio, è possibile utilizzare uno dei sottoinsiemi di Wikidata ottenuti mediante un tool chiamato WDumper, e disponibile qui: <https://wdumps.toolforge.org/dumps> (si suggerisce di considerare esempi di media grandezza, come ad es., Business, o Organization). Attenzione che con il pulsante next si accede a dumps ottenuti in precedenza; cliccando sul nome ci sono le specifiche usate per estrarre i dati. Il progetto sarà presentato mediante presentazione PPT in cui verranno discusse sia le regole implementate per ciascun pattern inferenziale, sia gli esempi ottenuti.